A sysadmin’s view of VoIP

Ewen McNeill
<ewen@naos.co.nz>
Naos Ltd

1. Introduction
   - Administriva
   - Legacy Phone System — A Review

2. Voice over IP
   - VoIP Protocols
   - Connecting to the PSTN
   - Challenges for the Sysadmin
   - Linux VoIP Software

3. Summary
Administrivia

About the speaker
- Runs Naos Ltd
- A Wellington based Linux, Unix and Networking consultancy

Informal Survey

Questions Policy
- If it is about the current slide, raise your hand.
- Please ask any other questions at the end.

Slides: http://www.naos.co.nz/talks/sysadmins-voip/
“Any sufficiently advanced technology is indistinguishable from magic.”
— Arthur C Clarke
- PSTN: Public Switched Telephone Network
- E.164: ITU standard for “phone numbers”
- DTMF: Dual-Tone Multi-Frequency “touch tones”
How the telephone system works — Part 3

- **PBX**: Private Branch Exchange
- Manages calls into and out of organisation
- Does phone number translation
- **ISDN**: Integrated Services Digital Network
- **BRI**: Basic Rate, 2 * 64Kbps data channels
- **PRI**: Primary Rate, 2Mbps (E1)
All VoIP operates in a similar manner:
- Control channel to set up call
- Media channels to carry encoded voice data
- Similar approach to FTP
- Lots of protocols for control and media channels
Control channel protocols

- H.323: ITU standard, uses ASN.1
- SIP: IETF RFC 2543, HTTP-like headers
- SCCP: “Skinny”: Cisco proprietary protocol
- Skype: Proprietary protocol based on Kazaa
- Several other less widely used protocols

SIP Example

INVITE sip:045551212@202.53.189.51;user=phone SIP/2.0
Via: SIP/2.0/UDP 161.29.192.202:5060
From: <sip:049714218@202.53.189.51;user=phone>;tag=1646388700
To: <sip:045551212@202.53.189.51;user=phone>
Call-ID: 3581559645@161.29.192.202
CSeq: 1 INVITE
Contact: <sip:049714218@161.29.192.202:5060;user=phone;transport=udp>
User-Agent: Cisco ATA 186 v2.15 ata18x (030313a)
Expires: 300
Content-Length: 256
Content-Type: application/sdp
Media channel protocols

- RTP: Realtime Transport Protocol
- RTP is ITU standard H.225.0
- And is also IETF RFC 1889
- Used by both H.323 and SIP
- Similar approaches used by other protocols
- Essentially timestamped UDP packets
- Between dynamically negotiated port numbers
Digitizing voice — codecs

- Same codecs used by H.323 and SIP
- All produce small packets: 50-250 data bytes
- G.7xx codecs are ITU standards:
  - G.711: 64kbps PCM (Pulse Code Modulation)
  - G.726: 16-40kbps ADPCM (Adaptive Differential PCM)
  - GSM: 13kbps, also used by GSM cellphones
- Codecs supported vary from product to product
- Patent and licensing issues around several codecs
Finding the other phone — Part 1

- Still need a way to locate the other phone
- Static configuration is possible — but doesn’t scale
- In H.323 a directory server is commonly used
- In SIP a proxy server can provide directory services via a redirection
Another common SIP proxy approach
Proxy in the middle of all control communication
Note how media channels still flow directly
ENUM: IETF RFC 3761: e164.arpa
Commonly proposed solution to finding the other phone
Being experimentally deployed at present
Encodes a E.164 (phone) number into a NAPTR DNS request
Take fully qualified number, reverse digits, separate by “.” (periods), and append .e164.arpa
+64-21-916-965 becomes 5.6.9.6.1.9.1.2.4.6.e164.arpa
Result of NAPTR query indicates protocol and location
VoIP hardware — Part 1

Many softphones (software on a PC) solutions exist
- For Linux options include:
  - GnomeMeeting: http://www.gnomemeeting.org/ (uses H.323)
  - KPhone: http://www.wirlab.net/kphone/ (uses SIP)
- Microsoft NetMeeting uses H.323
- Not as convenient as a “real phone” for most users
- Need a headset to get reasonable sound quality
  - Echo cancellation particularly a problem
- Wide range of hardware solutions available now
VoIP Hardware — Part 2

- Ethernet VoIP phone
  - Cisco, Uniden, etc
  - Protocols include SIP, H.323, “Skinny”

- Wireless (802.11b/g) VoIP phone
  - Often SIP only
  - Ability to roam through wireless network (eg Cafenet)
### VoIP hardware - Part 3

- **ATA: Analogue Telephone Adapter**
  - Cisco, Uniden, Sipura, etc
  - Usually 1-2 FXS (Foreign Exchange Station) ports for analogue handsets
  - Some include FXO (Foreign Exchange Office) ports to connect to the PSTN
  - Protocols include SIP, H.323, “Skinny” depending on vendor

- **VoIP Gateway**
  - Often modular, especially from router vendors
  - Accept FXO, FXS, BRI, PRI connections

- **Telephone line cards**
  - PCI (or ISA) cards for use in a PC
  - Combinations of FXO, FXS, BRI, PRI connections
  - Typically used in a PBX (or PBX replacement) situation
Connecting to the PSTN

- VoIP is useful, but limited, as a standalone system
- Ideally want to connect to legacy phone system
- To be able to make outgoing calls and receive incoming calls
- You can do this yourself (like a PBX)
- Or outsource it to a telco that accepts calls via VoIP and connects them to the PSTN (and vice versa)
Connecting to the PSTN yourself

- Requires a device that can talk VoIP on one side and to PSTN on the other
- And one or more suitable PSTN lines from a telco
- Solutions range from one simultaneous voice connection through dozens of simultaneous voice connections
- Depending on requirements (and budget!)
PSTN Interconnects

- PSTN interconnection can be analogue or digital
- Analogue connections suitable for home phone lines
- Connect PSTN to the FXO (Foreign eXchange Office) port on VoIP gateway or line card
- Business connections normally digital (ISDN)
- Either one or more BRI (Basic Rate) or PRI (Primary Rate) connection
- For a PRI you generally rent the circuit, plus as many 64Kbps timeslices as you need for simultaneous calls
Connecting to the PSTN typically requires phone number translation

Both incoming and outgoing

Similar to setting up a PBX

Incoming: direct calls to suitable VoIP phone (eg, receptionist)

Outgoing: put destination number in telcos preferred format

May need to strip off “outside line” prefix (eg, 1 or 9)

And possibly add a different prefix (eg, area code or country code)
You may want multiple connections to the PSTN
To save money (e.g., local calls to more numbers), or for reliability
Generally each interconnection will have its own phone numbers assigned to it
Translations and routing to cover all the connections can get quite complicated
May be easier to outsource to a provider that gives you good rates
Challenges for the Sysadmin

- Latency and jitter
- Faxes and VoIP
- Security
- Firewalls (and NAT)
VoIP is very sensitive to network performance
Including latency, jitter, and packet loss
Round trip latency of up to 300ms is okay; less is better
Significantly more sounds like a bad satellite phone call
Jitter (variation in latency) can be quite noticeable
Can be better to drop packets than delay them significantly
Occasional packet loss is tolerated by people
Latency and Jitter — Part 2

- Lots of spare bandwidth helps
- Prioritise VoIP traffic and/or reserve bandwidth for it
- For Linux see tc(8), and http://lartc.org/howto/lartc.qdisc.html
- Particularly a problem with low bandwidth (eg, residential) connections
- ADSL and wireless connections (in New Zealand) often have significant latency (100ms round trip)
- Cable modems are generally better (eg, 20-30 ms round trip)
- Upstream bandwidth is quite limited on many NZ ADSL connections (again cable modems are generally better)
Faxes and VoIP

- Sending faxes through a VoIP system can be challenging
- May be easier to persuade users to, eg, email a PDF
- Use a high bandwidth codec (eg, G.711 — 64kbps)
- May need to limit fax speed (eg, to 9600bps, V.29)
- Need a good network connection, and some luck
- T.38 Fax Relay is more reliable where supported
  - Effectively decodes fax locally
  - Then sends bitmap over network
  - And reencodes into fax tones at the remote end
By default both control channel and media channel are sent unencrypted

Encryption is thinly supported, especially in Open Source projects

For H.323, H.235 defines encryption (TLS or IPSec)

For SIP, TLS can be used

TLS: IETF RFC 2246: Transport Layer Security

SRTP: IETF RFC 3711: Secure RTP

May be best to run through a VPN for now, where possible

Or use a separate Layer 2 (VLAN or switch)
VoIP Security – Part 2

- Access to PSTN gateways needs to be secured
- And access to PBXes which are allowed to use those gateways
- Bouncing off company PBXes was a common phreaking technique to get “free” phone calls
- Security by static IP address ranges may be easiest to administer
- SIP includes a HTTP Basic-Auth style authentication
- But it is effectively in plain text
- H.323 also has a “shared secret” authentication scheme
VoIP and Firewalls

- VoIP control channel is usually a single well known port
  - H.323: TCP and UDP 1720
  - SIP: TCP and UDP 5060
- Other ports can be used as the port number is included in the protocol addresses
- Media channels are dynamically negotiated, often within a wide range of ports
- Assumes the “end to end” Internet
- Can lead to “one way audio”
Challenges of NAT

- Control channel can usually be NAT’d through firewall okay
- But media channel is challenging
- Because dynamic port negotiation includes IP addresses
- Meaningless outside the LAN if using RFC 1918 addresses
- Typical symptom is “one way audio”
- If both ends have the problem then no audio will be heard
- This is a moderately common issue with FTP as well, but there is better firewall support for FTP
Using a protocol aware firewall
- For Linux, sip-conntrack-nat: http://www.iptel.org/sipalg/
  (Alpha test code; in iptables Patch-o-Matic)
- For Linux, h323-conntrack-nat:
  http://max.kellermann.name/projects/netfilter/h323.html
  (Alpha test code; in iptables Patch-o-Matic)

Using an application level media proxy
- For Linux, Asterisk: http://www.asterisk.org/
- Or siproxd: http://siproxd.sourceforge.net/
- Or for H.323: OpenH323Proxy:
  http://openh323proxy.sourceforge.net/
Firewall and NAT solutions – Part 2

- STUN: IETF RFC 3489: Simple Traversal of UDP through NAT
- Tunnelling in unfiltered, globally unique, IP address
- Using vtun or GRE, or another VPN
- Will need to do policy routing to send traffic from those IP addresses back out the tunnel
- Linux: use iproute2 to route based on the source address range
- http://lartc.org/howto/lartc.rpdb.html
- Beware of security issues with tunneling in IP addresses

<table>
<thead>
<tr>
<th>Policy routing example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip route add default via REMOTEGATEWAY table TABLENAME</code></td>
</tr>
<tr>
<td><code>ip rule add from TUNNELEDBLOCK table TABLENAME</code></td>
</tr>
</tbody>
</table>

(Need to add TABLENAME to /etc/iproute2/rt_tables)
http://www.asterisk.org/

- Full functional, open source, PBX software for Linux
- Original developed by Digium Inc, who make PBX hardware including line cards (Zaptel)
- Supports SIP, IAX (Inter Asterisk eXchange), H.323 somewhat
- Huge range of features ("applications")
- All directly usable out of the dialplan ("extensions.conf")
Unfortunately configuration syntax combines the worst features of windows.ini and Basic

So you probably should not configure large dialplans by hand

Lots of front ends available now, eg

- Asterisk@Home: http://asteriskathome.sourceforge.net/
- Xorcom Rapid: http://xorcom.com/rapid

### Asterisk extensions.conf example

```plaintext
[nzpsten]
exten => _021[1-2]XXXXXX,1,Goto(outbound,${EXTEN},1)
exten => _021XXXXXX,1,Goto(outbound,${EXTEN},1)
exten => _02409XXXX,1,Goto(outbound,${EXTEN},1)

[congratulations]
exten => s,1,Wait,1 ; Wait a second, just for fun
exten => s,2,Answer ; Answer the line
exten => s,3,DigitTimeout,5 ; Set Digit Timeout to 5 seconds
exten => s,4,ResponseTimeout,10 ; Set Response Timeout to 10 seconds
```
Linux VoIP Software — SER

- SIP Express Router: http://www.iptel.org/ser/
- High performance SIP proxy
- Useful for routing SIP calls
- SIP registration support, database integration
- Configured in a C-like language
  - But beware the parser is fussier than C
- No media proxy support (only proxies control channel)

SER example

```c
if (method=="INVITE") {
    log(1, "Handling call INVITE");
    record_route(); /* Force subsequent requests here */

    if (!uri=~"sip:\+[0-9]+@.*") {
        log(1, "Rejecting non-E.164 destination");
        sl_send_reply("403", "Call cannot be served here");
        break;
    }
    ....
}
```
Other Linux VoIP Software

- GNU Bayonne:
  http://www.gnu.org/software/bayonne/bayonne.html
  - Telephony application server
  - Range of hardware and protocol support

- SIP-P: http://sipp.sourceforge.net/
  - SIP testing tool (protocol generator)

- Diagnostics
  - ngrep: http://ngrep.sourceforge.net/
  - ethereal: http://www.ethereal.com/
  - SIP is easier to debug than H.323, because you don’t need to decode ASN.1
That’s All Folks!

More information:
- http://www.voip-info.org: A useful Wiki, especially for Asterisk

Questions?

Slides: http://www.naos.co.nz/talks/sysadmins-voip/